Struct Multidisc Optim (2010) 42:619–631
DOI 10.1007/s00158-010-0521-y

RESEARCH PAPER

Feature-based spline optimization in CAD

A step towards geometry-based structure creation

Daniel Weiss

Received: 9 November 2009 / Revised: 31 March 2010 / Accepted: 15 May 2010 / Published online: 12 June 2010

Abstract In this paper, an advanced method for CAD-based spline structure optimization is investigated. The method is based on the combination of the commonly known parameter-based spline shape optimization and a recently presented feature-based structure variation concept for commercial CAD tools. The aim is to extend common parameter-based spline shape variation by the additional possibility to automatically add and remove control points or entire splines directly in CAD space. Such advanced spline modification provides a new level of flexibility in general geometry-based structural optimization. Using these splines to build CAD models, entirely new structures may be automatically generated during an optimization run through this newly gained flexibility in spline manipulation. The idea is to roughly define a continuous design space by basic splines and to gradually increase their shape complexity by control point number variation during optimization. Thus, operating on a knowledge-lean initialization—a design space bounded by basic splines and filled with material—this combination further extends the search and solution spaces of CAD-based structural optimization. The paper provides an outlook towards automated geometry-based structure *creation* combining nowadays commercial CAD software and a dedicated variation and optimization framework for geometry-based structural optimization.

Keywords Geometry-based structural optimization · Spline-based CAD variation · Knowledge-lean initialization · CAD specification tree

D. Weiss (✉)
ETH Zurich, Centre of Structure Technologies,
Leonhardstrasse 27, 8092 Zurich, Switzerland
e-mail: daniel.weiss@alumni.ethz.ch

List of acronyms

ASCII	American Standard Code for Information Interchange
BGL	Boost graph library
CAA	Component application architecture
CAD	Computer-aided design
CGM	CATIA geometric modeler
EA	Evolutionary algorithm
FE	Finite element(s)
FSS	Freestyle shaper
GSD	Generative shape design
NUPBS	Non-uniform polynomial B-splines
NURBS	Non-uniform rational B-splines
STEP	Standard for the exchange of product model data
TOD	Topology optimum design
UPBS	Uniform polynomial B-splines
VBA	Visual basic for applications

1 Introduction

Spline-based structural variation and optimization have long since been applied for continuous structure handling (Eschenauer et al. 1994; Jarraya et al. 2007; Albers et al. 2008; Brakhage and Lamby 2008) and reduction of design variables (Bös and Nordmann 2002). Apart from structural optimization through spline variation, the spline concept has also been used for FE-to-CAD transition of topologically optimized FE-structures (Koguchi and Kikuchi 2006; Jang et al. 2008). Leon et al. (2007) introduce the spline concept to FE-based topology optimization in combination with morphing. As all boundary FE nodes are kept, CAD spline control points can be directly linked to these nodes resulting in automatic CAD update when changing the FE nodes' positions. A similar approach using Fixed Grid FE-analyses was introduced by Garcia and Gonzalez (2004). Cervera and Trevelyan (2005a, b) describe a hybrid optimization approach in 2D and 3D applying FE-based topology optimization and using adaptive spline curves to keep outer structure and inner hole boundaries smooth.

The advantages of splines are therefore their smoothing and parameterization properties. Depending on their type (*interpolating* or *approximating*, see Fig. 1), their control points or the points of their according control polygon can be parameterized and allow for parameter-based shape variation (Albers et al. 2008). Due to their boundary properties (continuity in the second derivative), they grant smooth contours, frequently required by CAD models, and are therefore especially suitable for CAD-related structure variation (Fig. 2).

For purely spline-based topology optimization, Kim et al. (2008) apply closed B-spline curves for hole creation in addition to the boundary splines defining the structure's shape. Keller (2010) introduces a graph-based method applying spline and control point number variation for 2D Topology optimum design (TOD).

Apart from Keller's method, such spline-based structural optimization is usually based on a *fixed* number of splines and control points per spline. Hence, either a higher number of control points than possibly needed has to be defined prior to the optimization run or the number of necessary points has to be determined through dedicated reduction procedures (Brakhage and Lamby 2008).

By introducing such predefined splines and their *parameterized* control points into CAD, accordingly built CAD geometries can easily be varied and optimized applying the geometry-based parameter optimization method (Sprave et al. 2008). However, such parameter manipulation of predefined CAD structures only allows for very limited structure variation. Thus, the question arises if a more general concept for structure variation and even *creation* could be introduced in CAD.

1.1 Feature-based structure variation on CAD specification trees

A suggestion to tackle this issue can be found in the recently presented feature-based structure variation concept (Weiss et al. 2010). This concept additionally introduces topology variation, allowing to automatically instantiate and manipulate entire geometrical components of CAD models. Such extensive manipulation is performed through the access of the according specification trees—a hierarchical representation of the CAD geometry.

Based on these trees, which are provided by most commercial CAD tools, a dedicated representation and Evolutionary Algorithm optimization framework have been introduced to extend the state-of-the-art parameter-based shape and size optimization of CAD models. The concept additionally allows for *topology* variation directly on the highly constrained CAD models through the new possibility of automatically inserting and deleting geometrical subcomponents via the specification tree (Fig. 3). Thus, keeping the essential advantage of optimized ready-to-use CAD solutions, the search and solution spaces of CAD-based structural optimization are considerably extended.

The general idea is to transfer the CAD specification tree of an initial (prototype) CAD model into a directed mathematical graph. For each specification tree node, e.g. `var_Ribs_Add.1` in Fig. 3, a vertex is created and accordingly connected in the abstract graph (Fig. 4). The internal nodes represent *features*, i.e. geometrical components, and the final nodes are parameters. Such a graph can then be manipulated independently of any commercial CAD tool by according Evolutionary algorithm (EA) operators. These operators may perform tree branch (features and their sub-nodes) and leaf (parameters) *crossover*, as well as *mutation* of parameter values and the number

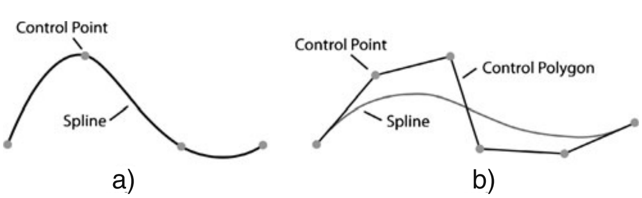


Fig. 1 Control point *interpolation* (**a**) and *approximation* (**b**)

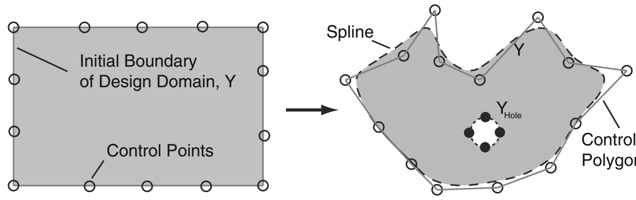


Fig. 2 Spline-based topology optimization (Kim et al. 2008)

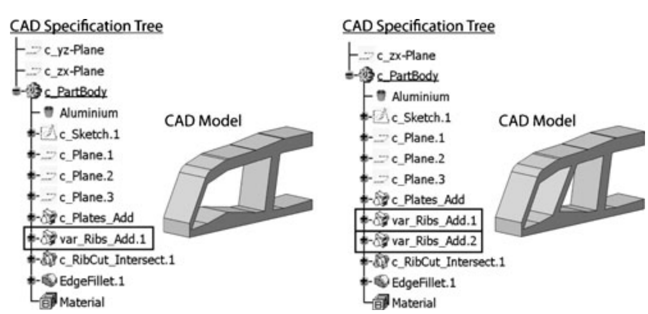


Fig. 3 Instantiating ribs as sub-components for topology variation

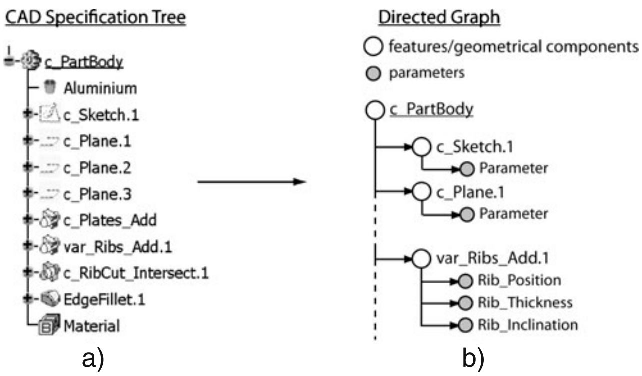


Fig. 4 Conversion of CAD specification tree (**a**) to directed graph (**b**)

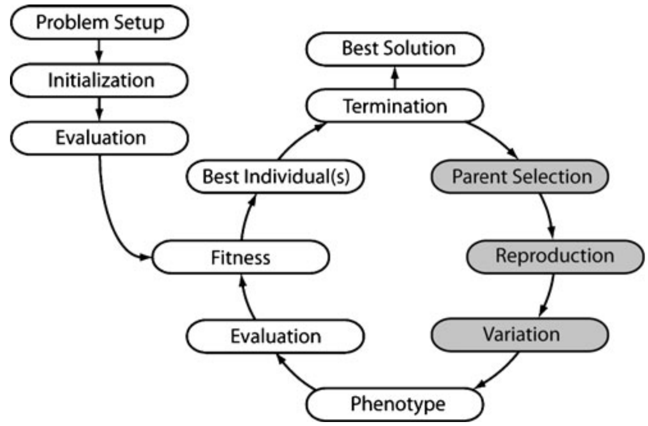


Fig. 5 Steps in an EA-based optimization loop (König 2004)

of variable features.[1] The operators are applied according to a certain probability to introduces a certain randomness in parameter and feature variation—similar to the random DNA mutations in nature. The mutation of the number of variable features virtually means to add/subtract corresponding nodes (including their sub-nodes) to/from the directed graph, e.g. `var_Ribs_Add.1` and its parameter sub-nodes in Fig. 4.

After manipulation, the graphs are transferred back to CAD space, i.e. the information is used to build the new according CAD structure (see Fig. 6). The rebuilding algorithm is CAD-software-dependent as it represents the interface between abstract graph and CAD space, i.e. it has to directly access the CAD information. In the current case (for CATIA V5), these routines have been implemented in CATIA's programming interfaces, Visual Basic for Applications (VBA) and the Component application architecture (CAA) RADE. For further details, the reader is referred to Weiss et al. (2010).

Evolutionary algorithms For the sake of completeness, a short description of the tailored optimization algorithm operating on these specific directed graphs is given. The algorithm is based on an EA as such algorithms have been proven to be suitable for CAD-based structural optimization problems (König 2004; Ledermann 2006; Giger 2006). The EA performs the typical steps of *selection*, *reproduction* (crossover), and *variation* (mutation) on a set of representations of CAD candidate solutions (Fig. 5)—the previously mentioned mathematical graphs.

However, the operated representations in this case are non-standard hierarchical trees, i.e. the directed graphs, representing a CAD model. These trees contain real and discrete values (parameters represented by tree leafs) as well as geometrical features (entire sub-trees) to be varied. Therefore, the optimization algorithm operators perform exchange and mutation operations on these nodes, distinguishing between such different node types. The created candidate solutions are directed graphs that are subsequently used to rebuilt the new CAD model, which is then assessed by FE-simulations (Algorithm 1, Fig. 6).

Generally, the new possibilities of combined parameter and topology variation pose further challenges regarding the efficiency of the optimization algorithm. Through the varying numbers of components in the representation, the algorithm needs to cover a larger search space, which may reduce the optimization run's convergence efficiency. This issue has been addressed by Weiss et al. (2010) and will not be elaborated in detail here. The basic idea was the introduction of a timed shift from topology to parameter variation during the run, i.e. shifting from exploration to exploitation, and thus gradually narrowing the search space. Such adaptation resulted in considerable improvements of the necessary population sizes and average result quality compared to an algorithm without such an adaptation.

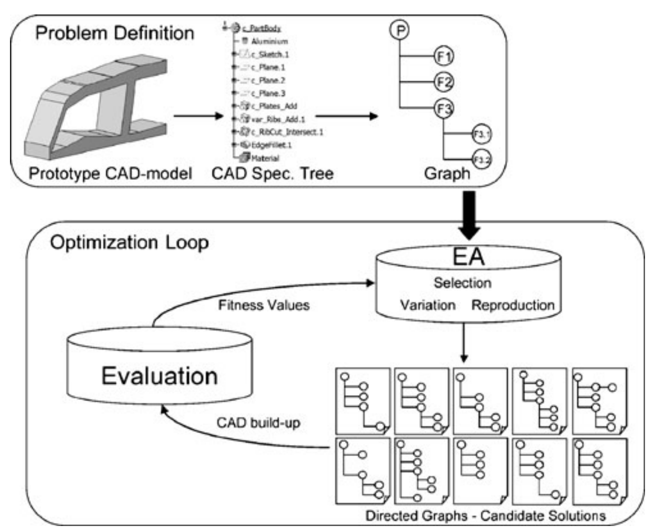


Fig. 6 Schematic display—optimization on hierarchical CAD representations

[1]The EA operators in this case were implemented in C^{++} using the Boost graph library (BGL) for efficient manipulation of such mathematical graphs; http://www.boost.org/.

Algorithm 1 EA-based opt. on specification trees

Convert CAD prototype to abstract graph $D^{proto} \rightarrow G^{proto}$
Initialize population (through mutation m_p applied to graph prototype) $\mathbf{G}(0) \leftarrow N \cdot (m_p \cdot G^{proto})$
for each new individual g in $\mathbf{G}(0)$ **do**
  Convert graph representing the individual to CAD geometry $g \rightarrow D^{new}$
  Evaluate individual through FE-analyses: Fitness $\mathcal{F}(g)$
end for
Get best individuals $\mathbf{G}(0)_{best} \subset \mathbf{G}(0)$
while not max. number of generations/iterations reached **do**
  Create offspring through crossover (single nodes or sub-trees) between pair of best individuals (graphs) of previous generation $t-1$
  $\frac{N}{2} \cdot ((g_i \in \mathbf{G}(t-1)_{best}) \cap (g_j \in \mathbf{G}(t-1)_{best})) \rightarrow \mathbf{G}(t)$
  for each new individual g in $\mathbf{G}(t)$ **do**
    Vary new offspring through mutation m of parameter values and number of variable sub-components with probability p: $g' \leftarrow m_p \cdot g$
    Convert individual to CAD $g' \rightarrow D^{new}$
    Evaluate individual: Fitness $\mathcal{F}(g')$
  end for
  Get best individuals $\mathbf{G}(t)_{best} \subset (\mathbf{G}(t) \cup \mathbf{G}(t-1)_{best})$
end while
End optimization loop

A further general issue regarding the newly introduced variation of geometrical features, e.g. ribs, holes, in CAD is the remaining requirement of predefined and possibly highly constrained prototypes of such features. This means that, even though the shape and size of newly inserted sub-components are adapted during the optimization run, the user has to provide a rough initial CAD prototype of these variable components. Thus, although considerably extending *geometry-based* structural optimization by topology variation, the *creation* of entirely new structures is hardly possible.

Regarding this issue, the more flexible spline structures may provide a possibility to achieve higher flexibility in geometry-based structure optimization and creation. The previously used knowledge-rich initializations, i.e. the predefined prototypes, may therefore be substituted by more general spline structures.

The combination of these spline structures in a further step with the feature-based concept to insert and remove geometrical components, e.g. spline control points, may then finally yield an extensive possibility for comprehensive structure variation. In other terms, the initial parameter-based spline variation is considerably extended (Subsection 1.1).

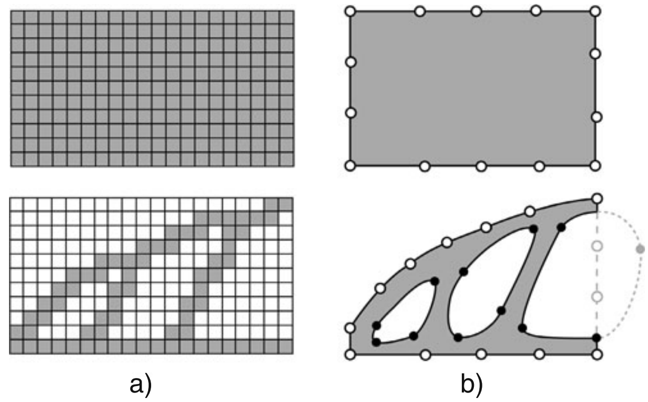


Fig. 7 Scheme of FE-based (**a**) and spline-based (**b**) topology variation

The work this paper is based on, is motivated by the flexibility of spline-based shape optimization and the new possibilities of CAD-based structural variation that may allow for increased solution space coverage due to less *knowledge-rich* initializations. The purpose is not to introduce entirely new concepts specifically for spline-based structural optimization as such methods have been thoroughly investigated over the last few years (Eschenauer et al. 1994; Bös and Nordmann 2002; Kim et al. 2008). Equally, this concept does not try to substitute the FE-based method (Bendsœ and Kikuchi 1988), Fig. 7a, in any way. The structure *creation* ability of the FE method using *knowledge-lean* initializations and thus working on a much less constrained solution space is still unmatched by any geometry-based approach, which operates on comparatively heavily constrained CAD models.

The focus is rather set on the combination of spline-based continuous geometries, commercial CAD software, and the extended recently introduced CAD-based topology variation concept for comprehensive CAD spline optimization (including control point and spline number variation). The resulting findings and framework may provide a further step towards extensive and fully automated structural optimization in commercial CAD tools and industrial standard processes with the essential benefit of optimized ready-to-use CAD models.

2 Control point feature variation

Introducing the spline concept into CAD tools, the shape and topology of such a CAD geometry are rendered considerably more flexible. Instead of assembling knowledge-rich—and thus highly constrained—variable features to a

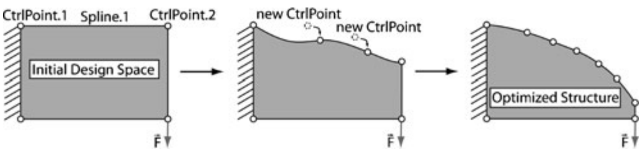


Fig. 8 Initial design space and adapted space through control point feature insertion

final geometry (Section 1.1), the inner and outer boundaries of the initial design space are varied before filling it with isotropic material (Fig. 2). Furthermore, with the possibility of feature number variation, i.e. inserting additional spline control points, no knowledge about the suitable number of control points is required. Thus, the amount of necessary knowledge for the initial structure is considerably reduced, which further extends the search and solution spaces for geometry-based structure variation up to actually allowing for structure *creation*.

The idea is to use commonly available parametric interpolation splines of commercial CAD tools[2] to define the boundaries of the continuous initial design space in CAD and to fill the space with a homogeneous material. Such an initial design space is nothing else than the continuous counterpart to the discrete space used in FE-based topology optimization (Bendsœ and Kikuchi 1988; Kim et al. 2000). However, instead of deleting finite elements or reducing their density, the design space boundaries are adapted. Hence, a *geometry-based* instead of an *FE-based* structural optimization method. In addition, holes, created by closed splines, may be inserted to alter the topology of the filled design space (Fig. 7).

For an optimization problem setup, the initial design space is simply defined by corner points marking the maximum extension of the initial search space. Two neighboring corner points are linked by a spline resulting in linear boundary segments (Fig. 8, left). Hence, only the starting and ending points of the splines are set without any intermediate control points. Such points are later inserted at random positions during optimization using the *feature-based* structural variation concept via the specification tree (Subsection 1.1). Each such control point is a *variable feature* that can be inserted (or deleted) to allow for a varying spline shape complexity. As a result, a more and more developed and adapted spline structure is created (Fig. 8).

2.1 Splines in CAD

The spline control points are provided as separate parametric features, which alleviates their access via the specification tree. They are simple points in 3D space, defining the spline curve (Fig. 9).

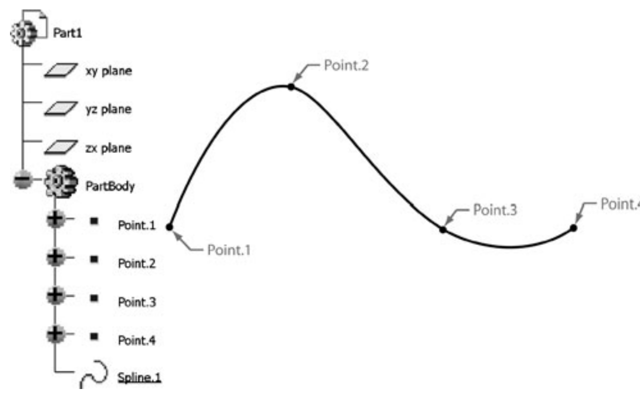


Fig. 9 Splines in CAD (CATIA V5) and their specification tree representation

The 3D splines used in this project are cubic or quintic curves of interpolating type, depending on whether additional tangency constraints are applied or not. A more detailed description of general spline types and spline distinction specifically in CATIA V5 can be found in (McKinley and Levine 1998; Rogers 2001; Brill 2006).

According to the idea of feature-based spline point variation (Fig. 8), CAD-based shape variation can then be performed as shown in Fig. 10—adding/subtracting control points as variable features.

However, although providing a suitable specification tree representation, such spline features bear a major issue concerning curve control. Due to the fact that the curves are of *interpolating* type and thus do not provide a *control polygon* for manipulation, the curve behavior on interpolation point variation is rather difficult to predict. Too close points or too abrupt direction changes may result in unexpected curve behavior yielding invalid (Fig. 11) or unnecessarily complex CAD structures. For this reason, a careful setting of the point's position parameters is required.

The following section, therefore, presents an additional concept to increase parameter variation control and thus to reduce unexpected curve behavior on control point insertion or variation during the optimization process.

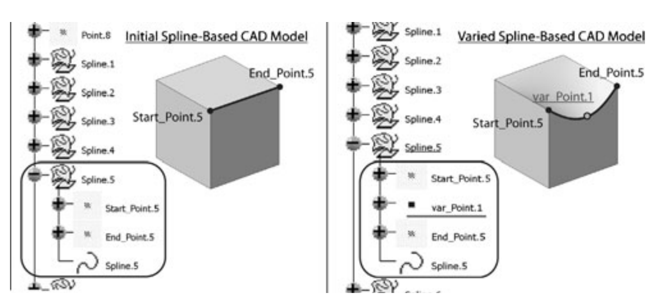


Fig. 10 Spline-based shape variation in CAD (CATIA V5)

[2] In the present work, Dassault Systèmes' CATIA V5 (http://www.3ds.com)—a wide-spread commercial CAD tool in automotive and aerospace industry—was used as representative commercial CAD tool.

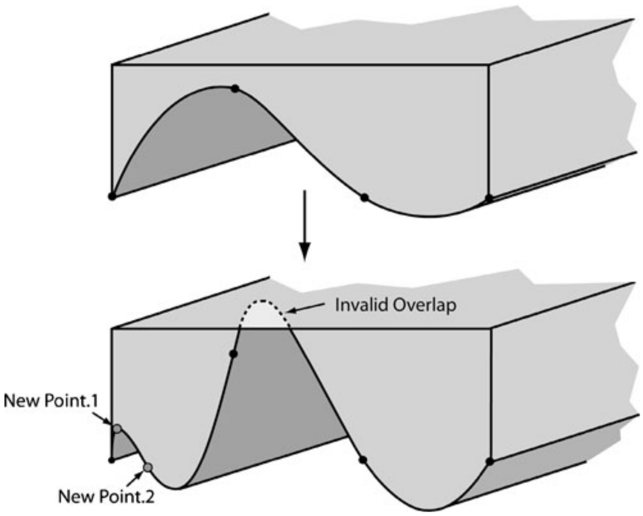


Fig. 11 Spline-based solid becoming invalid due to insertion of new spline control points

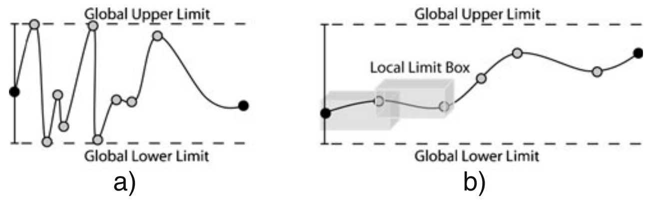


Fig. 12 Global parameter limits (**a**) and improved curve behavior through local sequential limits (**b**)

3 Constraining spline curve behavior

The curve behavior of *interpolating* splines heavily depends on the position of neighboring control points relative to each other. Always considering its continuity in the second derivative, the curve may result in unexpected curve shapes due to badly placed control points. Such behavior may lead to invalid CAD structures and does not meet the *causality* criterion (small changes in the genotype should lead to small changes in the phenotype) for efficient optimization.

3.1 Point position variation control

To increase causality on control point variation or insertion, a dedicated variation method for the point's positioning parameters is required. Hence, a concept is needed to roughly consider the neighborhood of the current point to be varied, i.e. its neighboring interpolation points, to keep the varied position values close to the values of the neighboring points. As a result, considering the already existing points' positions when varying the actual point, would favor a smooth curve behavior.

The required optimization algorithm, thus, has to be able to *sequentially* vary these interpolation points, always considering the previous, i.e. the already varied points, to restrict the variation limits for the new location of the next point[3] (Fig. 12).

Lee et al. (2007) introduce an approach using a *Move Ranges* definition for each control point. The ranges are based on a predefined formula including the distance to the neighboring control points and thus defining dependent upper and lower parameter limits for the according coordinate variation range. The actual concept is roughly based on this approach, however, considering the additional possibility to add and remove control points, further dedicated constraint handling is required.

Sequential dynamic parameter limits Prior to the variation of the position coordinates, the upper and lower limits of the ranges in which the x, y, and/or z coordinates are to be varied are adapted. This adaptation is done considering the already set position coordinates of the preceding control point. Thus, the coordinates of the previous point are taken as relative references for the local ranges in which the coordinates of the actual point may be varied (Fig. 12b).

The sequence of the control points is defined by a reference direction/parameter marked by the user, e.g. the x-coordinate. Each time the control point position values of a candidate solution (the graph representing a CAD geometry) are subjected to crossover or mutation, the sequence is newly calculated and the local position ranges are adapted step by step.

After adapting the range of the actual point, the parameter value itself may then be varied between the new limits through the EA-operators (e.g. mutation). Applying this procedure, global limits, i.e. the global maximum and minimum values, of a parameter can be defined as well as the dynamic and sequentially dependent local limits (Fig. 13).

Algorithm 2 gives an exemplary overview over local limit adaptation and subsequent parameter mutation after a new generation of candidate solutions has been generated (see row 9 and following in Algorithm 1). The control points

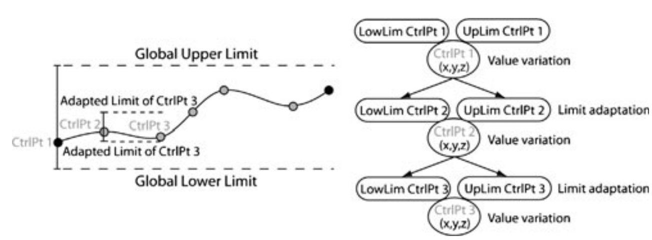


Fig. 13 Local limits defined by previous control point and sequential parameter variation

[3]The mutation operator of the Evolutionary Algorithm may then vary the point's position between the newly adapted upper and lower parameter limits.

Algorithm 2 Dynamic local parameter limits

...
while not max. number of generations/iterations reached **do**
  Create offspring through crossover $(g_i \in \mathbf{G}(t-1)_{best}) \cap (g_j \in \mathbf{G}(t-1)_{best}) \rightarrow \mathbf{G}(t)$
  for each new individual g in $\mathbf{G}(t)$ **do**
    Get sequence of control points according to position x $\rightarrow \mathbf{C}$
    for each c in $\mathbf{C}$ **do**
      Adapt upper/lower limits of y, z coordinates: $y'_{minNewPt}, y'_{maxNewPt} \leftarrow y_{prevPt} \pm \delta y$ $z'_{minNewPt}, z'_{maxNewPt} \leftarrow z_{prevPt} \pm \delta z$
      Mutate y, z coordinates between the adapted ranges with probability p: $c(x, y', z') \leftarrow m_p \cdot c(x, y, z)$ with $y'_{minNewPt} \leq y' \leq y'_{maxNewPt}$ $z'_{minNewPt} \leq z' \leq z'_{maxNewPt}$
    end for
    Convert individual to CAD $g' \rightarrow D^{new}$
    Evaluate individual: Fitness $\mathcal{F}(g')$
  end for
end while
End optimization loop

are sequential in x direction and allow variations in y and z direction.

The definition of these global and local ranges is done during the optimization problem setup using according formulas or fix values.

Having added further curve behavior control through sequential local parameter limit restriction, according optimization runs in CAD can be performed.

In the following section, two basic setups for advanced spline-based shape variation and multi-level structure creation will be presented. The examples provide basic findings concerning behavior and main issues of such advanced geometry-based structural optimization.

4 Study cases

The presented study cases in this section are run on very basic and academical CAD structures. The aim is to investigate optimization run and structural behavior, applicability, and effectiveness of geometry-based structural optimization using CAD spline curves. According findings will provide a basis for further research and application of this highly experimental approach towards geometry-based structure *creation*. The used commercial CAD software is CATIA V5.

4.1 Single-level volume optimization

The aim of this study case was to investigate the ability of geometry-based feature variation to operate on comparatively knowledge-lean initializations of 3D solid structures. Although actually not being *topology* but rather advanced spline *shape* optimization, the current case is suitable to investigate the method's extended capabilities for structural variation using CAD spline features.

The CAD prototype is a continuous design space volume created by cubic spline curves, which define the according faces bounding the volume. The volume is the continuous counterpart to a discretized cubic design space used in FE-based topology optimization. Four of the spline edges contain an according variable control point prototype to be instantiated N times. They are instantiated at random positions between start and end points of their spline during optimization to allow for more complex spline shapes than the initial straight lines in the prototype.

The CAD model is single-leveled regarding the levels of variable features: only the number of control points is to be varied on each spline, the number of splines is kept constant. To reduce calculation time, x- and y-axis symmetries were considered, resulting in a quarter of the final structure. As further symmetry can be used on this quarter, only half of the quarter was subjected to spline point variation (Fig. 14).

Because the analysis workbench of CATIA V5 only allows for load and restraint application on static boundary representations of the CAD geometry (faces, edges), the according areas are not to be modified. CATIA uses non-static naming of boundary representations and therefore changes the name, i.e. the reference used by the load and restraint components, upon each change of the face. Thus, static face patches had to be defined for the top force application area, the bottom fixation, and the lateral sliders substituting the rest of the symmetric structure (Fig. 15).

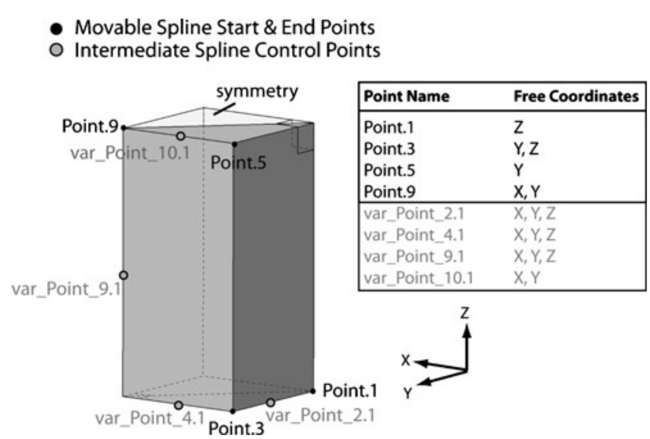


Fig. 14 Cube: CAD prototype, variable control points and free parameters

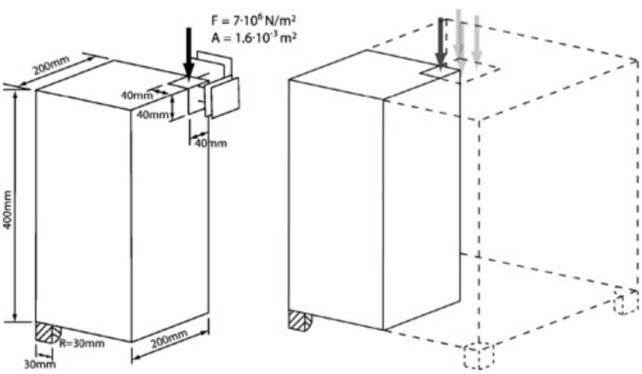


Fig. 15 Cube: load case and static facial patches

To control curve behavior (Section 3), the global and local upper and lower limits of the control point positions had to be carefully defined. Using the newly added sequential parameter consideration, the intermediate, i.e. the variable, control points are kept roughly between the start and end point positions with slight variations (Fig. 16).

The optimization run was then performed applying minimum deformation energy as design objective and a mass constraint to 10% of the initial design space. The fitness value $\Phi(\mathbf{p})$ of each candidate solution is the weighted sum of constraints and design objectives—a cumulative fitness function. There was no buckling considered to simplify the optimization run.

$$\Phi(\mathbf{p}) = w_e D_e(O') + w_m D_m(C) \tag{1}$$

p is the phenotype vector—in this case, an n-dimensional abstract vector containing parameters and features. $D_e(O')$, $D_m(C)$ are normalized objective/limit rating functions for deformation energy (D_e) and mass (D_m) according to König (2004), see Fig. 17. w_e, w_m are manually chosen weighting coefficients ($0 \leq w \leq 1$) to provide a balanced optimization path between the competing optimization targets *mass* and *deformation energy*.

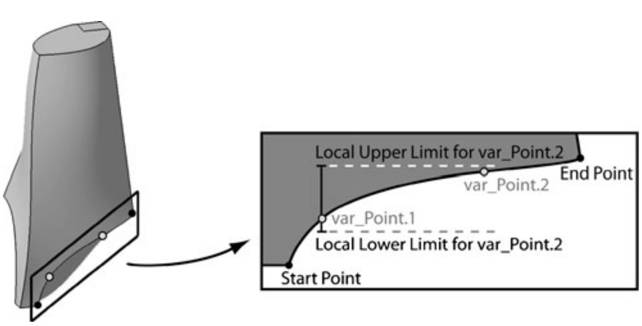


Fig. 16 Local limit definitions and resulting CAD spline

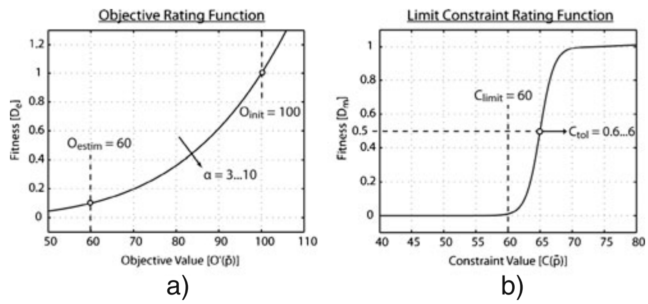


Fig. 17 Exemplary rating functions for design objectives (**a**) and constraints (**b**)

With $D_e(O')$ for the deformation energy *design objective*:

$$D_e(O') = (aO' + b)^5 \tag{2}$$

O' = current deformation energy (FE-analysis)

$$a = \frac{1 - \sqrt[5]{0.1}}{O'_{init} - O'_{estim}}$$

$$b = 1 - aO'_{init} \tag{3}$$

O'_{init} = average initial deformation energy

O'_{estim} = estimated optimized deformation energy

and $D_m(C)$ for the mass *limit constraint*:

$$D_m(C) = \frac{1}{1 + e^{-\lambda(C - C_{limit} - \Delta)}} \tag{4}$$

C = current mass

C_{limit} = maximum allowed mass

$$\lambda = \frac{1}{C_{tol}}\left(ln\left(\frac{1}{D(C_{limit})} - 1\right) - ln\left(\frac{1}{D(C_{tol})} - 1\right)\right)$$

$$\Delta = \frac{1}{\lambda} ln\left(\frac{1}{D(C_{limit})} - 1\right) \tag{5}$$

C_{tol} = tolerated deviation from maximum mass

As shown in Figs. 18 and 19, the structure development is mainly driven by the aim to reduce the mass from 100% to the 10% fraction of the initial design space. However, the design objective (keeping the deformation energy at a minimum) still successfully guides the shape towards optimum

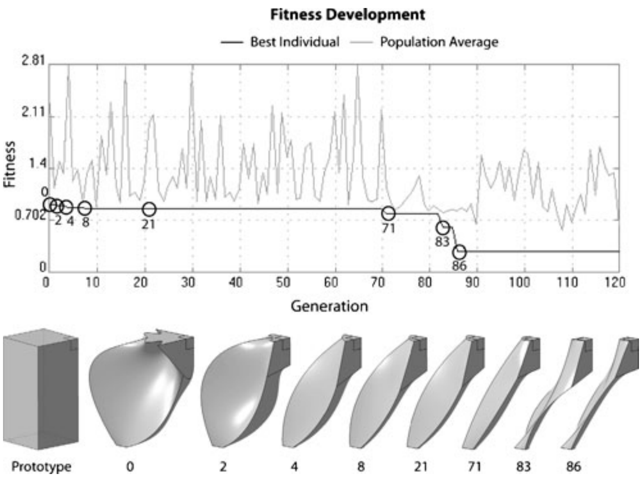


Fig. 18 Cube: optimized CAD structure and fitness development

force flow and thus drives the resulting structure towards the roughly expected shape for the applied load case (Fig. 15). Due to the rather simple structure, few interpolation points had to be inserted to adapt the boundary splines—generally one point per spline.

A particularly interesting development can be observed between energy, mass and fitness development from the beginning of the optimization run to generations *70* to *86* (Figs. 18 and 19). While considerable mass changes are required at the beginning to compensate the penalty of energy increase, the large energy peak from generation *82* to *83* is easily compensated by a relatively moderate mass reduction. This behavior clearly reflects the characteristics of the applied fitness rating function for constraints (Fig. 17). The function introduced by König (2004), and slightly modified for the present studies, is designed to operate on candidate solutions of the Evolutionary Algorithm yielding constraint values within a reasonable range of the optimum value. If the values lie far outside such a range, the gradient is rather small. This means for the actual case that, starting at a point far outside the allowed *mass* value range of $C_{limit} = 0.1 * 43.2$ kg $= 4.32$ kg, the mass rating

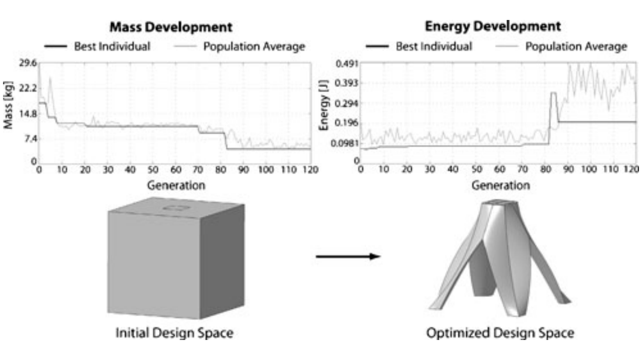


Fig. 19 Cube: mass and energy development during structural optimization

function has only a slight gradient to indicate the direction towards the valid range. Hence, if the mass is lowered, the increase in the *energy* objective penalty has to be compensated by a large *mass* reduction. Thus, the EA continuously approaches the valid area of the mass limit constraint by loosing considerable weight and keeping energy increase at a minimum—the fitness changes only slightly. Finally, at generation *83*, the EA has reached the valid range where the mass penalty gradient is much higher. This means that a small step towards C_{limit} already compensates the penalty of a large energy increase. Hence the peak, which is rapidly mitigated again through further optimization from generation *85* to *86*.

An according suggestion towards higher convergence efficiency through a more suitable constraint rating function is provided by Giger (2006). Through the increase of the weight factor of a constraint violating a limit, the optimization algorithm temporarily receives a stronger focus on this constraint. For the present case, this means that also small mass reductions may compensate temporary energy increases due to a possibly higher penalty weight of the mass constraint.

Generally, although an acceptable result was achieved, the optimization on such a spline structure revealed further issues. Despite the application of spline point variation control (Section 3), still a high rate of infeasible individuals (40–50% of the population size) could be observed due to the mentioned low causality between control point variation and curve behavior. This high sensitivity can be found again in large fluctuations of the average population fitness (Fig. 18). Furthermore, in the present case, the FE-solver of CATIA V5 tends to remain in eternal mesh adaptation loops if heavily distorted geometry is submitted to FE-analyses, which requires several interruptions and restarts of the optimization run.

Nevertheless, with the achieved optimum structure, the applicability of the introduced geometry-based optimization concept to comparatively *knowledge-poor* initializations could be proven. The optimization is obviously able to add as many additional interpolation points as needed to achieve the optimum spline shapes. Thus, *geometry-based* advanced spline shape optimization could be successfully introduced into commercial CAD tools.

This concept may be further extended by the introduction of variable hole features, allowing for additional *topology* optimization. These hole features may be basic cylindrical shapes or even closed splines (see Fig. 2). Their number and position can be varied during optimization again using the feature-based topology variation concept. However, although such topology variation on the *knowledge-poor* initialization of the 3D cube would be the final step towards geometry-based structure *creation*, this subject will not be further addressed here. Instead, the focus is set on the investigation of interpolation point *and* spline number variation—a general multi-level issue not only arising when inserting spline holes but also when generally extending the degrees of freedom in shape optimization.

The next subsection will therefore investigate this multi-level issue on a very basic sheet geometry, including basic topology variation through simple hole insertion.

4.2 Multi-level sheet variation

The multi-level case was set up as an extended investigative example aiming towards structure variation using a variable number of control points *and* splines. Recalling the need of an already highly developed CAD prototype base structure for geometry-based structure variation, this section investigates the possibility to apply CAD-based variation much earlier in the design process—using less developed CAD structures. Hence, the actual academical load case is dedicated to the automated adaptation of a sheet to specific constraints, applying two levels of variable features. The number of splines defining the face may be varied as well as their number of intermediate control points. This means that, during optimization, mutation operators may randomly add or remove splines. Newly inserted splines (consisting solely of a start and an end point) are placed at a random position in y-direction (see prototype spline in Fig. 20a). Then, on these new and on existing splines, additional control points may be inserted—also randomly—between the according start and end points of each spline.

A variable number of holes was additionally allowed to provide a possibility for material removal. Figure 20 shows the CAD prototype and the applied load case. The geometry consists of a main half where the features and parameters are varied and its mirrored counterpart (shaded, Fig. 20a).

The aluminum plate is loaded with a facial distribution force of $26.59\frac{N}{m^2}$, simulating its weight and an additional linear force of $19.62\frac{N}{m}$. The design objectives were chosen to be again deformation energy and mass minimization (see (1)) without any stress constraints to keep the optimization problem as simple as possible (faster convergence due to less trade-offs between constraints). As the only aim was to investigate the new concept's ability to respond to specific constraints and objectives, no global optimization of the plate was performed, which would have required the

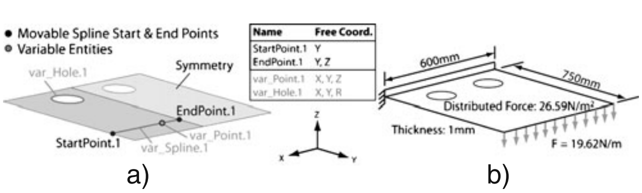


Fig. 20 CAD prototype (**a**) and load case (**b**) for sheet structure optimization

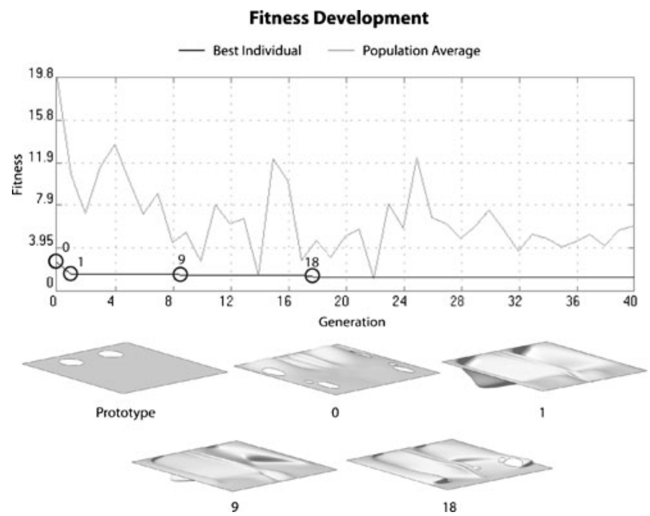


Fig. 21 Sheet: optimized CAD structure and fitness development

additional consideration of yield stresses and buckling constraints. To additionally consider maximum yield stresses—or any additional constraint—an according limit constraint (D_σ) could simply be included into the cumulative fitness function of (1):

$$\Phi(\mathbf{p}) = w_e D_e(O') + w_m D_m(C) + w_\sigma D_\sigma(C) + \ldots \qquad (6)$$

In Figs. 21 and 22, the results of an exemplary run are shown. The algorithm initially favors deformation energy reduction until a certain threshold is reached. In the subsequent iterations, a Pareto-frontier between minimum mass and energy is searched, slowly lowering both values. The resulting geometry—after several generations without further improvement—is obviously not (easily) manufacturable. This clearly shows the trade-off between *knowledge-rich* and *knowledge-poor* initializations. Allowing for greater flexibility in structural variation usually implies less control over manufacturability and validity of the optimized structure.

Concluding this section, two study cases were investigated regarding the advanced *shape* and basic *topology* creation capabilities of geometry-based structure variation

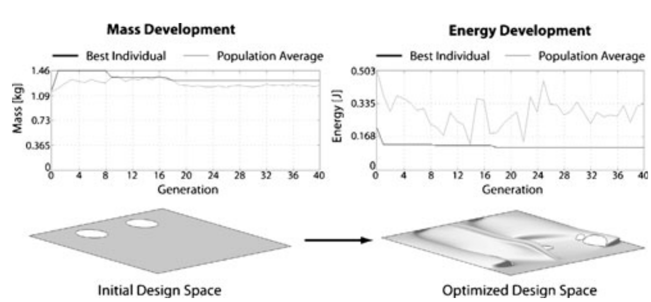


Fig. 22 Sheet: mass and energy development during structural optimization

using spline curves for *continuous* design space definition. The cases proved the advanced shape optimization ability of CAD-based structural variation on knowledge-lean initializations as a further step towards geometry-based *structure creation*. However, the concept implies several issues mainly resulting from the use of interpolating instead of approximating splines. This leads to a curve behavior that is difficult to control and extremely high rates of invalid CAD structures, which are increased if solid structures are built from Non-uniform rational B-splines (NURBS) surfaces, resulting in further unpredictable behavior of the geometry.

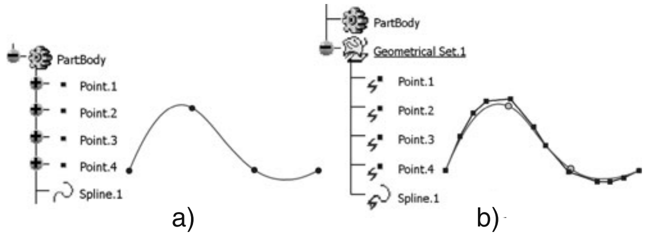


Fig. 24 Parametric geometry (**a**) and imported spline curve with non-parametric points

5 Alternative CAD spline control

Considering the findings of the previous section, it would be neglectful not to provide further thoughts about alternative spline curve variation possibilities in CAD. This section, therefore, provides according ideas for further research as a starting point for projects involving CAD-related spline-based structure variation.

5.1 ASCII-based control polygon variation

As the main issue in CAD is frequently the lack of a parameterized control polygon to encapsulate the curve and prevent unexpected curve shapes, the definition of such a polygon might be of first priority.

If the CAD software is not able to display such a polygon, i.e. its parameterized points, in the specification tree, there is a work-around via non-parametric geometry. Examining the spline description in a STEP file, the parametric interpolation spline is converted into a B-spline with an according control polygon. If the STEP file is again imported, the former *spline control points* become non-parametric simple points lying on the curve, without any linkage anymore to the curve itself (Figs. 23, 24, 25).

Hence, focusing on the created control polygon in the STEP file, the according polygon point coordinates may be altered through simple parameter variation. Such altered files can be again imported in CAD (Fig. 26), to create the modified curve. The initial parametric interpolation points are not used anymore.

Such *parameter* variation may be done by any available EA or other variation algorithms. The approach may even be extended to the variation of the number of polygon points, simply inserting or deleting according ASCII lines in the STEP file for additional points. A dedicated file format handling would then be necessary to grant valid STEP files to be imported into CAD. Such an approach, however, is not anymore based on specification trees (Subsection 1.1) and requires an accordingly adapted optimization algorithm. Nevertheless, it may be a considerably more robust alternative to the parametric *spline control point* variation.

5.2 Spline variation via mathematical descriptions

A further possibility for more robust spline variation may be given through the extraction of the mathematical descriptions of the CAD curves. In CATIA V5, this may be done through the CAA RADE interface. The interface grants access to the core structures of the CATIA geometric modeler (CGM)—amongst others, to the underlying mathematical descriptions of curves and splines. This approach again differs from the feature-based concept introduced in this work and requires an accordingly tailored optimization algorithm.

Using classes such as `CATSplineCurve`, `CATNurbsCurve`, and `CATKnotVector`, the user may retrieve all needed information to mathematically describe the actual

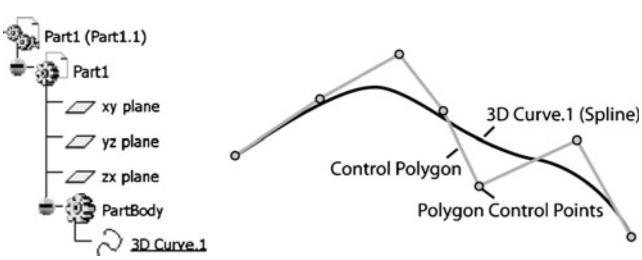
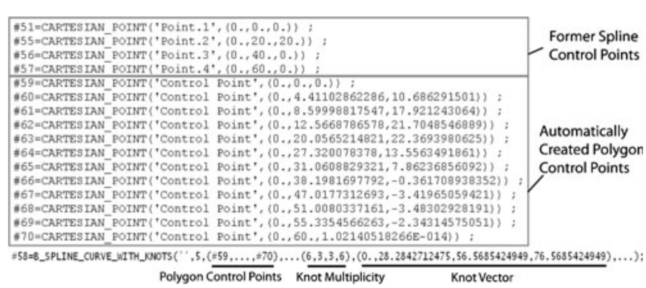


Fig. 23 Example for (temporary) control polygon in CAD (CATIA V5 FSS workbench)



```
#51=CARTESIAN_POINT('Point.1',(0.,0.,0.)) ;
#55=CARTESIAN_POINT('Point.2',(0.,20.,20.)) ;
#56=CARTESIAN_POINT('Point.3',(0.,40.,0.)) ;
#57=CARTESIAN_POINT('Point.4',(0.,60.,0.)) ;
#59=CARTESIAN_POINT('Control Point',(0.,0.,0.)) ;
#60=CARTESIAN_POINT('Control Point',(0.,4.41102862286,10.686291501)) ;
#61=CARTESIAN_POINT('Control Point',(0.,8.59998817547,17.921243064)) ;
#62=CARTESIAN_POINT('Control Point',(0.,12.5668786578,21.7048546889)) ;
#63=CARTESIAN_POINT('Control Point',(0.,20.0565214821,22.3693980625)) ;
#64=CARTESIAN_POINT('Control Point',(0.,27.320078378,13.5563491861)) ;
#65=CARTESIAN_POINT('Control Point',(0.,31.0608829321,7.86236856092)) ;
#66=CARTESIAN_POINT('Control Point',(0.,38.1981697792,-0.361708938352)) ;
#67=CARTESIAN_POINT('Control Point',(0.,47.0177312693,-3.41965059421)) ;
#68=CARTESIAN_POINT('Control Point',(0.,51.0080337161,-3.48302928191)) ;
#69=CARTESIAN_POINT('Control Point',(0.,55.3354566263,-2.34314575051)) ;
#70=CARTESIAN_POINT('Control Point',(0.,60.,1.02140518266E-014)) ;
#58=B_SPLINE_CURVE_WITH_KNOTS('',5,(#59,...,#70),...(6,3,3,6),(0.,28.2842712475,56.5685424949,76.5685424949),...);
```


Fig. 25 Spline description in STEP—including automatically generated control polygon points

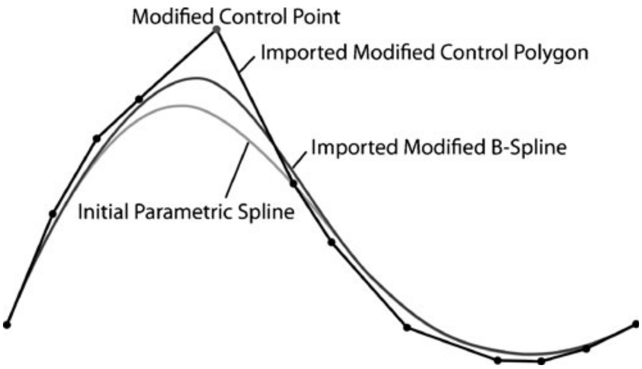


Fig. 26 Robust curve shape change due to polygon point variation

curve. As a result, shape variation can be performed directly via the mathematical equations, which may be easier to monitor.

However, for this approach, the rather expensive CAA RADE environment is needed as well as sound knowledge of C^{++} and CAA. Nevertheless, the CGM offers a vast variety of interfaces for direct geometrical manipulation and information retrieval.

Such access to mathematical descriptions of geometrical CAD objects may even provide a possibility to link CAD with current structural variation methods carried out on rather mathematical than CAD models (Jarraya et al. 2007; Keller 2010).

6 Summary and conclusions

In this paper, investigations concerning advanced geometry-based shape variation on *knowledge-lean* initializations in CAD were conducted providing a further step towards geometry-based structure *creation*. With the introduction of spline-based CAD models, considerably less restricted structural variation than using predefined and highly constrained variable features could be performed.

In nowadays commercial CAD tools, the spline curve and its interpolated control points are represented in the specification tree, which alleviates their access and manipulation. Varying their number through the recently introduced feature variation concept allows to gradually increase the initial spline structure's complexity to create entirely new CAD structures. The optimization results of the described study cases prove the general applicability of geometry-based structural variation using such *knowledge-lean* initializations and commercial CAD tools.

However, if there is no parameterized control polygon and the interpolation points have to be directly manipulated, the spline curve behavior may become highly unpredictable. The curve is not constrained by the convex hull of the polygon and may thus result in unexpected shapes to meet its continuity in the second derivative.

Through the addition of dynamic sequential parameter limits to define locally dependent ranges for parameter variation, curve control could be enhanced. Nevertheless, especially in volumetric structures, defined by spline face boundaries, the rate of invalid candidate solutions remains extremely high due to additional facial intersections. As the faces are automatically created between the spline curves, too strong curvature may result in even heavier distorted surfaces intersecting with other neighboring facial boundaries of the volumetric structure.

A further issue results from the rather uncommon and possibly inappropriate shapes generated through spline variation. Meshing algorithms may remain in eternal mesh adaptation loops for too narrow or small areas in the volumetric body.

More suitable approaches for CAD spline curve variation may be the concepts outlined in Section 5 serving as a basis for further research projects.

Concluding, the combination of the feature-based variation concept and spline structures provides a considerable step towards geometry-based structural *creation*. Even though further research is needed for comprehensive application of this concept, it provides a new level of flexibility for geometry-based structure optimization on CAD models.

Acknowledgments I would like to thank Daimler AG for the funding and providing the necessary infrastructure at the Daimler Research Center in Ulm, Germany. In addition, I greatly appreciate the support and supervision of Prof. Dr. Paolo Ermanni and the Centre of Structure Technologies at the Federal Institute of Technology (ETH) in Zurich, Switzerland.

References

Albers A, Rovira NL, Aguyo H, Maier T (2008) Optimization with genetic algorithms and splines as a way for computer aided innovation. International Federation for Information Processing (IFIP) 277:7–18. doi:10.1007/978-0-387-09697-1

Bendsœ M, Kikuchi N (1988) Generating optimal topologies in structural design using a homogenization method. Comput Methods Appl Mech Eng 71:197–224

Bös J, Nordmann R (2002) Reduction of structure borne sound by numerical optimization. In: Forum acusticum, revista de acustica, vol 23, pp 1–6

Brakhage KH, Lamby P (2008) Application of B-spline techniques to the modeling of airplane wings and numerical grid generation. Comput Aided Geom Des 25:738–750

Brill M (2006) Parametrische Konstruktion mit CATIA V5—Methoden und Strategien für den Fahrzeugbau. Carl Hanser Verlag München, Wien

Cervera E, Trevelyan J (2005a) Evolutionary structural optimization based on boundary representation of NURBS. Part I: 2D algorithms. Comput Struct (Elsevier) 83:1902–1916. doi:10.1016/j.compstruc.2005.02.016

Cervera E, Trevelyan J (2005b) Evolutionary structural optimization based on boundary representation of NURBS. Part II: 3D algorithms. Comput Struct (Elsevier) 83:1917–1929

Eschenauer HA, Kobelev VV, Schumacher A (1994) Bubble method for topology and shape optimization of structures. Struct Optim 8:42–51

Garcia MJ, Gonzalez CA (2004) Shape optimization of continuum structures via evolution strategies and fixed grid finite element analysis. Struct Multidisc Optim 26:92–98. doi:10.1007/s00158-003-0323-6

Giger M (2006) Representation concepts in evolutionary algorithm-based structural optimization. PhD thesis, ETH Zurich

Jang GW, Kim KJ, Kim YY (2008) Integrated topology and shape optimization software for compliant MEMS mechanism design. Adv Eng Softw 39:1–14

Jarraya A, Dammak F, Abid S, Haddar M (2007) Shape and thickness optimization performance of a beam structure by sequential quadratic programming method. J Fail Anal and Preven 7: 50–55

Keller D (2010) Evolutionary design of laminated composite structures. PhD thesis, ETH Zurich

Kim DH, Lee SB, Kwank BM, Kim HG, Lowther DA (2008) Smooth boundary topology optimization for electrostatic problems through the combination of shape and topological design sensitivities. IEEE Trans Magn 44(6):1002–1005

Kim H, Querin QM, Steven GP, Xie YM (2000) A method for varying the number of cavities in an optimized topology using evolutionary structural optimization. Struct Multidisc Optim 19:140–147

Koguchi A, Kikuchi N (2006) A surface reconstruction algorithm for topology optimization. Eng Comput (Springer) 22:1–10. doi:10.1007/s00366-006-0023-0

König O (2004) Evolutionary design optimization: tools and applications. PhD thesis, ETHZ

Ledermann C (2006) Parametric associative CAE methods in preliminary aircraft design. PhD thesis, ETH Zurich

Lee S, Kwak BM, Kim IY (2007) Smooth boundary topology optimization using B-spline and hole generation. Int J CAD/CAM 7:16–32

Leon N, Cueva J, Villarreal C, Hutron S, Campero G (2007) Automatic shape variations for optimization and innovation. International Federation for Information Processing (IFIP) 250:179–188

McKinley S, Levine M (1998) Cubic spline interpolation. Online, http://online.redwoods.cc.ca.us/instruct/darnold/laproj/Fall98/SkyMeg/Proj.PDF

Rogers DF (2001) An introduction to NURBS with historical perspective. Morgan Kaufmann Publishers

Sprave J, Lengl E, Teutsch R, Krumenaker T, Nowottny D (2008) Optimierung auf der Basis von parametrischen CAD-Modellen unter Berücksichtigung von Betriebsfestigkeitsanforderungen. In: DVM (ed) Optimierungspotenziale in der Betriebsfestigkeit, Betriebsfestigkeit, vol DVM-Bericht 135, pp 59–68

Weiss D, Sprave J, Kress G (2010) Introducing geometry-based topology optimization supported by cad specification trees. Submitted to Computer-Aided Design